

04

## Mostrar uma conta

Vamos começar o formulário de alteração de uma conta. Esse é um pouco mais trabalhoso, então vamos desenvolver juntos.

Vamos começar pela `Action` que mostrará uma conta, a partir de um ID. Ela recebe um ID, busca no banco essa `Conta`, e manda pra JSP:

```
@RequestMapping("/mostraConta")
public String mostra(Long id, Model model) {
    ContaDAO dao = new ContaDAO();
    model.addAttribute("conta", dao.buscaPorId(id));
    return "conta/mostra";
}
```

Agora vamos para essa JSP. Ela será um formulário, mas a diferença é que esse formulário já trará os dados preenchidos. Repare também no campo invisível com o ID. Isso é importante para que consigamos saber qual conta será alterada. Veja:

```
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<html>
<body>
    <h3>Alterar Conta</h3>
    <form action="alteraConta" method="post">
        Descrição: <br/>
        <textarea name="descricao" rows="5" cols="100" ${conta.descricao}></textarea>
        <br/>
        Valor: <br/>
        <input type="text" name="valor" value="${conta.valor}" /><br/>
        Tipo: <br/>
        <select name="tipo">
            <option value="ENTRADA" ${conta.tipo=='ENTRADA' ? 'selected':''}>Entrada</option>
            <option value="SAIDA" ${conta.tipo=='SAIDA' ? 'selected':''}>Saída</option>
        </select>
        <br/>
        Pago? <input type="checkbox" name="paga" ${conta.paga?'checked':''} />
        <br/>
        Data de Pagamento: <input type="text" name="dataPagamento" value=<fmt:formatDate value:>
        <br/>
        <input type="hidden" name="id" value="${conta.id}" />
        <input type="submit" value="Alterar"/>
    </form>
</body>
</html>
```

Para o Spring MVC saber converter automaticamente a data no formato brasileiro para um `Calendar` é preciso usar a anotação `@DateTimeFormat`. Abra a classe `Conta` e adicione a anotação acima do atributo `dataPagamento`:

```
@DateTimeFormat(pattern="dd/MM/yyyy")
private Calendar dataPagamento;
```

Falta criar um método que cuidará da alteração da conta. Na classe `ContaController` adicione esse método:

```
@RequestMapping("/alteraConta")
public String altera(Conta conta) {
    ContaDAO dao = new ContaDAO();
    dao.altera(conta);
    return "redirect:listaContas";
}
```

Faltou somente o link na listagem:

```
<td><a href="mostraConta?id=${conta.id}">Alterar</a></td>
```

E pronto. Teste. Tudo funcionou.

Esse código é bastante extenso e complicado, mas é assim que fazemos alterações. O que achou? O que podemos melhorar nesse código?

## Responda

INserir Código		Formatação